Timing Picking Waves in a Warehouse

Kevin R. Gue
Department of Systems Management
Naval Postgraduate School
Monterey, CA 93943
kevin.gue@nps.navy.mil

January 24, 2001

Abstract

To facilitate efficient picking tours and worker supervision, warehouses often release orders to workers in picking *waves*. The timing of these waves determines how long orders sit in the queue waiting to be processed, and so provides some control over average order cycle time. We show how to time picking waves in a warehouse to minimize average order cycle time. We report on results at a warehouse in California, which reduced average order cycle time by more than 25% after restructuring its picking waves. We also argue that cycle time may not be the best metric for many distributors, and propose a different one based on the practices of a leading e-commerce distributor.

Keywords Warehousing, Distribution, Measurement and methodology, Personnel and shift scheduling

1 Background

Reducing cycle time in the supply chain means lower inventories and better response to customers. As distribution firms wring the last drops of inefficiency out of their distribution channels, warehouses have focused on tightening up processes and reducing costs. Moreover, the need to satisfy increasingly demanding customers has led to new performance measures that will require even *more* emphasis on process optimization and cost cutting.

In the future, firms will have to adapt to the world of e-commerce distribution: direct customer orders via the Internet, precise order tracking, stringent performance measurement, and even auction-based services. As many e-commerce retailers have found, customer satisfaction goes beyond the web experience to effective distribution and order delivery. The Internet has also bred the impatient customer, who expects delivery of products ordered on the web almost as quickly as the clicks it took to place the order. Firms are responding by restructuring processes to reduce order cycle time in the warehouse. The thought is that reduced order cycle time will improve customer service.

In the warehouse, one component of order cycle time is the time an order spends waiting to be downloaded, or pulled, from the information system. Waiting time occurs because work is not released to workers continuously, but rather in batches called picking *waves*. Warehouses routinely use picking waves to improve the efficiency of picking tours. When orders are batched, the fixed time that it takes to circumnavigate a picking area is amortized over more orders, thus reducing average cycle time; but if the batches are too large, cycle time goes up because orders wait excessively in the queue.

We investigate the effects of the number and timing of picking waves on average order cycle time at a warehouse. We describe a simple model, which we solve with a genetic algorithm, that finds the optimal times to release work to the warehouse, given its specific order arrival distribution. We report the use of the model at a large warehouse in California, which reduced average order cycle time by more than 25% after restructuring its picking waves. After presenting the results, we argue that minimizing cycle time in most cases is *not* the right objective. We present a superior objective and show how to modify our model to accommodate it.

2 Choosing picking waves

The test site for our study is a large, multi-warehouse complex (hereafter, "the warehouse") located in California, which supports customers throughout the United States and Asia. The warehouse handles all types of material, from pallet loads to individual piece parts, and employs several methods of storage and material handling, including block stacking, pallet rack, man-aboard picking cranes, and mini-stackers.

Because the warehouse supports customers from such a geographically diverse region, orders arrive throughout the 24-hour day. Customers order via an information system that holds orders in a queue until the warehouse downloads, or *pulls*, them from the system. Once the orders are in the warehouse system, they are transmitted to handheld or mounted RF devices maintained by the workers, creating a *wave* of orders. Workers then traverse their areas, picking orders into totes or onto pallets as appropriate. From the picking area, orders move to packing, then on to the shipping area, where workers sort them by customer and transportation mode.

The number and timing of picking waves determine to some degree the cycle time for orders, because they determine how long orders spend in the queue. The size of the wave also determines how long a picking tour will be, which also affects cycle time.

2.1 Theory and practice

Why not release the work continuously? In theory, releasing the work in a continuous stream minimizes average order cycle time. Imagine that a new order arrives and is transmitted to a worker in an aisle at the moment he is about to pass the item's location.

Proposition 1 *Passing over a pick always increases average order cycle time.*

Proof Assume total travel in the aisle takes non-zero time s and there are $n+1$ picks remaining in an aisle, each of which takes time t. Assume, without loss of generality, that we are considering whether or not to pick the next item in the aisle. Consider the average cycle time delay experienced by all $n+1$ items: if we pick the next item, then total delay is t for each of n items, and average delay is $\frac{nt}{n+1}$; if we pass the next item and pick it on the next tour, then total delay is at least $s + nt$ for that item, and average cycle time delay is $\frac{s+nt}{n+1} > \frac{nt}{n+1}$; therefore, average cycle time is greater if the pick is passed over. $\square$

Proposition 1 suggests that the optimal order picking strategy is to have workers circumnavigate the warehouse continuously, picking orders as they appear on their handheld RF devices. In practice, this is infeasible for a number of reasons. First, order pickers typically have other duties involving paperwork, cleaning, and stowing that are normally done between picking waves. The wave structure allows supervisors to assign workers to other tasks before the next wave begins. Second, the warehouse management system must support the continuous feed, inserting and sorting orders appropriately as they arrive. (The warehouse management system at our test site was not configured to do this.) Third, workers could spend much more time traveling, especially during slow periods, thus wasting labor; that is, continuous picking minimizes cycle time, but at the expense of higher labor costs.

What is the minimum time between waves? The answer depends on a number of things: the time required to circumnavigate each picking area, the time to pick an individual order, the order arrival pattern, and the amount of slack time needed by supervisors to accomplish other things with the workers. At the warehouse, managers suggested that picking waves separated by 2 hours were possible, but waves in consecutive hours were not.

2.2 The model

Consider a warehouse that receives orders continuously, but not necessarily uniformly, throughout every 24-hour cycle. Every hour the warehouse has an opportunity to release all orders in the queue to the warehouse, or it may delay until a future hour.

Assume that orders arrive to the warehouse every hour according to some distribution. We seek a set of n picking waves that minimizes average order cycle time. Because total system throughput (orders/day) is fixed, we know by Little's Law that minimizing the average work in process inventory also minimizes the average cycle time.

Planning picking waves is similar to the well-studied production lot-sizing problem with time-varying demand (see Nahmias, 1997, for a discussion of different solution methodologies). In our case, inventory holding cost is simply one hour per unit per time period, but the notion of setup cost is different. In the lot-sizing problem, one incurs a monetary setup cost each time a new lot is ordered. In our problem the warehouse incurs a fixed time penalty for each wave (in the form of a picking tour) that could prevent scheduling a wave in the next hour. We work around this issue by specifying the number of waves a priori.

We use the following notation:

Indices

t denotes time periods,

Sets

T denotes the circularly ordered set of time periods,

Data

a_t the average number of orders that arrive in period t,

Decision variables

X_t equals 1 if orders are released to pickers in period t, and 0 otherwise, and

I_t WIP inventory in the queue at the beginning of period t.

We order the elements of T circularly, so that hour 0000 follows hour 2300.

The objective is to

$$\begin{aligned}
\text{Minimize} \quad & \sum_{t \in T} I_t & & \\
\text{subject to} \quad & I_{t-1}(1 - X_{t-1}) + a_t = I_t & \forall\, t > 0 \qquad & (1)\\
& I_t \geq 0 & \forall\, t \qquad & (2)\\
& X_t \in \{0, 1\} & \forall\, t. \qquad & (3)
\end{aligned}$$

The objective is to minimize average work in process inventory, which we know by Little's Law will give us the picking waves that minimize average cycle time. Constraint set 1 is the material balance equation. Notice that this is a non-linear, integer constraint. Constraint sets 2 and 3 define appropriate variables as non-negative and binary.

We implicitly assume that minimizing queue time is equivalent to minimizing cycle time. Our intution had been that the size of a wave would affect processing time, but data from the warehouse suggested otherwise: we found no correlation between wave size and processing time for orders in the batch.

Although the problem is relatively small, it is not easy to solve to optimality because of the non-linear, integer constraint. We use a simple spreadsheet model and an embedded genetic algorithm (see Michalewicz, 1996, for an introduction). We use the EVOLVER 4.0 solver from Palisade Corporation (1998) as an add-in to EXCEL, with default values for crossover and mutation rates. (The relatively small size of our problem — and quick solution times — make these parameters unimportant.) The user specifies the required number of waves, and the solver chooses the times that minimize average WIP inventory in the queue. Solutions to our model generally take less than a couple of minutes.

3 Results at the test site

We obtained data on order arrivals to the warehouse, analyzed the current picking waves, and used the model to recommend new picking waves. We analyzed arrival and processing data for the months September through November, 1999. The data set included more than 100,000 orders.

Figure 1 shows the order arrival stream. Managers speculated that the large peak around 0500 was probably due to order releases from East Coast customers early in the morning. The warehouse operated staggered shifts, with the earliest workers arriving at 0400 to "catch the first wave."

[Figure 1 about here.]

Before we began working with them, the warehouse had recently reduced the number of waves from three to two, one at 0300, the other at 1100 (see the arrows in Figure 1). Because it sometimes takes more than 20 minutes to download the orders in a wave, the 0300 wave was timed to meet workers on the first shift at 0400. Managers and supervisors said they believed workers were "more efficient" with fewer, larger waves, because the density of picks is greater. Interestingly, managers had never looked at the hourly arrival patterns.

Figure 2 is a histogram of queue times for orders in September, 1999. There are two peaks, one at 6 hours, and one at 15 hours. The peak at 6 hours corresponds to the time between the arrival peak at 0500 and the 1100 wave, while the 15 hour peak corresponds to the time between the 1200 peak in arrivals and the 0300 wave the following morning. These peaks are also evident in the total cycle time data, although they are mitigated somewhat by processing time variance (see Figure 3).

[Figure 2 about here.]

[Figure 3 about here.]

In November, the warehouse began experimenting with more picking waves, eventually settling on 4 waves, at 0400, 0600, 1200, and 1600. These times correspond very closely with the set chosen by our model. Figures 4 and 5 show the dramatic shift in queue times and cycle times for requisitions in November.

[Figure 4 about here.]

[Figure 5 about here.]

Figure 6 illustrates the average queue time for orders for different numbers of picking waves. The figure shows the marginal decrease in average queue time with each additional wave: beyond 4-5 waves, each additional wave reduces queue time only a matter of minutes. At the test site, managers settled on 6 waves, because it was enough to give them low cycle times, but not so many that waves were being released before the previous waves were completed. Managers began experimenting in November, and settled on 6 waves around December or January. The waves are at 0400, 0600, 0900, 1100, 1400, 1600. The waves recommended by the model were identical, with the exception of the middle two, which the model recommended at 1000 and 1200. (Managers preferred the earlier times to better match existing break schedules.) Figure 7 shows the average cycle times before and after changing the wave structure. Average cycle time for January–April is 25% lower than that from August–October.

[Figure 6 about here.]

[Figure 7 about here.]

4 Performance metrics and customer service

A reality of distribution is that customers rarely care how efficiently a warehouse receives or stores its inventory; they want their goods as quickly as possible and in good condition. To know how well a warehouse is performing the shipping function, managers construct *metrics*, which are most often numerical representations of data collected in the warehouse.

4.1 Average days delayed

Prior to our study, managers at the test site recorded performance with a metric called *average days delayed*, which is the average number of days required to ship an item, where each observation is rounded down. For example, an item that ships on the same day it arrived has zero days delayed; an item that ships anytime during the next day has one day delayed, and so on. For this measure, they defined a *day* by the clock; thus an item arriving at 0500 arrives the same day as one arriving at 2300, although it has a much greater chance of shipping the same day (and achieving zero days delayed).

Note that to say that an item "has shipped" can have different meanings, depending on the transportation mode. For example, if the warehouse uses common carriers such as UPS or FedEx, "shipped" may mean that it is ready for pickup, even though the carrier will not arrive for several hours. For a warehouse that controls its own fleet of trucks, "shipped" may mean that the truck has left the dock. For our purposes, we use the first definition; that is, we consider an item to have shipped when it is available for transporting.

The intent behind average days delayed is to push workers to ship items on the same day they are received, thus giving customers the advantage of receiving their orders sooner.

As a performance metric, average days delayed has at least three problems:

- The performance of the warehouse depends directly on the arrival times for orders, making it difficult to compare performance of different warehouses using the same metric. For example, a warehouse having a large influx of orders late in the day will necessarily perform poorly compared to one having a large influx earlier in the day. This is an important point, because one of the primary uses of metrics by managers is to compare different warehouses in the firm.

- The measure could cause supervisors to unnecessarily schedule irregular work hours, thus increasing labor costs. For example, if a large number of orders typically arrives late in the day, managers might schedule a late shift at higher labor cost per hour. But unless the late shift allows those orders to be transported on an earlier route, the customer sees no advantage.

- Improving average days delayed does not necessarily improve customer service. For example, if there is a large surge late in the day, managers might bring in a large afternoon shift to process all those orders before midnight, thus reducing average days delayed; but if those orders still ship the next day, there is no advantage to the customer.

Just before we began our work, executives at the test firm decided to change to an average order cycle time metric. They believed that the increased resolution of this metric would more accurately reflect actual warehouse performance.

4.2 Average order cycle time

The cycle time of an order is the difference between the time it arrives at the warehouse and the time it is ready for shipping. The average cycle time metric records the average value for all orders.

Average cycle time is less affected by the timing of order arrivals than average days delayed. Two warehouses having a large influx of orders but at different times of the day should fare the same in average cycle time, although a later surge may require higher cost of labor.

Unfortunately, average order cycle time can still cause managers to construct perverse workforce schedules. For example, if a surge of orders arrives overnight, there is an incentive to bring workers in during the mid-shift to get those orders processed, even though the orders cannot be shipped until the FedEx pickup late that afternoon. Those orders might have been processed just as well during normal work hours. Improving average order cycle time does not necessarily improve customer service. Reducing the average cycle time from 8 to 7 hours will have no effect unless some orders actually ship earlier.

4.3 A better metric

Average days delayed and average order cycle time fail to necessarily improve customer service because they are *internally-focused* metrics. They measure to different degrees how long it takes an order to flow through the warehouse, without regard to how soon the customer sees his order. In fact, both measures *tend to* improve customer service, but only as a side effect.

These metrics ignore the larger context of how a supply chain responds to customer orders. Warehouses and manufacturing sites can respond to demand continuously, because orders flow from the facility in a stream. Transportation is different because of the need for economies of scale—orders are served in batches, such as truckloads, whether they are constructed by the warehouse or by the carrier. Batching necessarily introduces a periodic nature to the distribution system, and it is this batching that the two measures ignore.

To establish a *customer-focused* performance metric, we must account for the batching that occurs in transportation. We proposed to the test site a *Percent making Cut-Off* (PCO) metric that records the fraction of orders arriving before an established cut-off time that makes the next shipment cycle.

For example, suppose a carrier such as UPS or FedEx makes a pickup at 1700 every day, and we establish a cut-off time of 1400. The PCO metric would record the fraction of orders arriving before 1400 every day that ship at 1700 the same day. The idea is to ship as many orders as possible on the same day. This is the same goal aimed at by average days delayed, but here we have more sensibly defined "day".

The PCO metric has the advantage of driving rational workforce scheduling. It gives managers the incentive to plan in such a way that as much work as possible is out of the way prior to the cut-off time. Because the last minute rush happens at the same time (or times) every day, managers can free up resources to handle the increased load.

Also, improvements on the PCO metric result in better customer service by definition. Moreover, cut-off times provide the warehouse with an effective marketing tool: Because customer service is fundamentally about meeting or exceeding customer expectations, firms are at a disadvantage when the customer does not know what to expect or expects more than the firm can deliver. The PCO metric allows the warehouse to publish the cut-off time, clarifying expectations and improving customer satisfaction. For example, if a customer submits an order one hour after the cut-off time, he has little or no expectation that his order will ship that day. If the warehouse happens to get the order out, the customer's expectation is exceeded; if not, it is met.

Internet software retailer PC Connection provides an excellent example of this strategy in practice. PC Connection publishes on its web page that orders received before 0200 will be received the next day (or the same day, in the case of orders between midnight and 0200) PC Connection (2000). The customer's expection is clearly defined, and the warehouse has a measureable, customer-focused goal each night. (We do not know if their warehouse uses such a goal.)

4.4 Modifying the model

To optimize picking waves with respect to the PCO metric, we modify our model slightly: First, identify the latest pickup time for each delivery mode (FedEx, UPS, LTL carrier, etc.), and estimate the lead time required to process a batch of requisitions for those modes. Next, add a constraint, $X_t = 1$, for the appropriate hour, and solve the model. The solution will optimize around that fixed wave.

Implementing the PCO metric in our test site would be complicated by the fact that they use several shipping modes, with different cut-off times. Also, their information system would have to be modified to sort orders and create appropriate waves. At the time of this writing, executives at the test firm were considering the metric and its implications for operations.

For any firm, implementing the PCO metric requires careful analysis and execution. For example, arriving orders would have to be sorted according to their transportation modes and released at an appropriate time, possibly not according to the first come, first served discipline. To set cutoff times, warehouses would have to estimate processing times accurately.

5 Conclusions

Warehouses can reduce average order cycle time by properly scheduling their picking waves. In general, waves should be released shortly after peaks in the arrival distribution to ensure that the largest group of arrivals waits for the least amount of time.

Restructuring picking waves according to our model led to a 25% reduction in average order cycle time at the test site. It was interesting to note during the study that managers believed that fewer waves meant more efficient picking tours and therefore better performance, when in fact, it meant just the opposite. Increasing the number of waves reduces time spent in the queue, and, for the warehouse, larger waves did not necessarily reduce processing time.

Our results suggest that 4-6 waves are most appropriate for data similar to that of our test site. With more waves, there is a potential for confusion, as waves could be released before work is complete on the previous wave. With fewer waves, orders tend to spend too long in the queue.

We also contend that minimizing average order cycle time is not the right metric for warehouses using cyclical transportation providers, because it ignores the periodic nature of pickups. We propose a *Percent making Cut-Off* (PCO) metric that establishes a cut-off time for orders to be guaranteed shipping on the next delivery cycle. The metric correctly aligns incentives inside the warehouse to maximize performance to the customer. It also manages a customer's satisfaction by managing his expectation: customers ordering after the cut-off time have no expectation of receiving their orders in the next cycle.

References

Michalewicz, Z., 1996. Genetic Algorithms + Data Structures = Evolution Programs. Springer, Berlin.

Nahmias, S., 1997. Production and Operations Analysis. Irwin, Chicago, p. 82.

Palisade Corporation, 1998. Guide to Evolver.

PC Connection, Jun. 2000. Web site `www.pcconnection.com`.

List of Figures

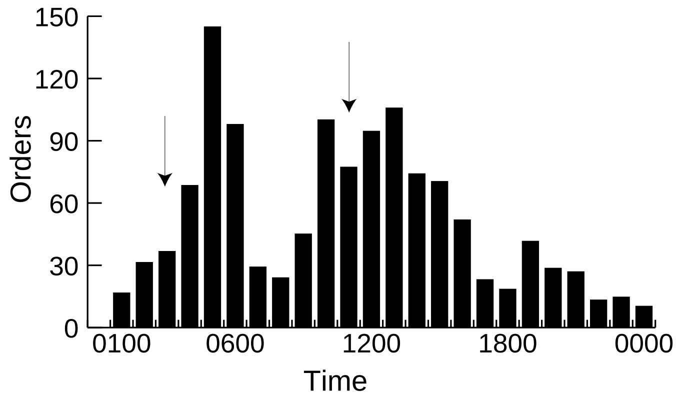

Figure 1: Average hourly order arrivals during September–November, 1999 for the test site. Arrows indicate the times of the picking waves.

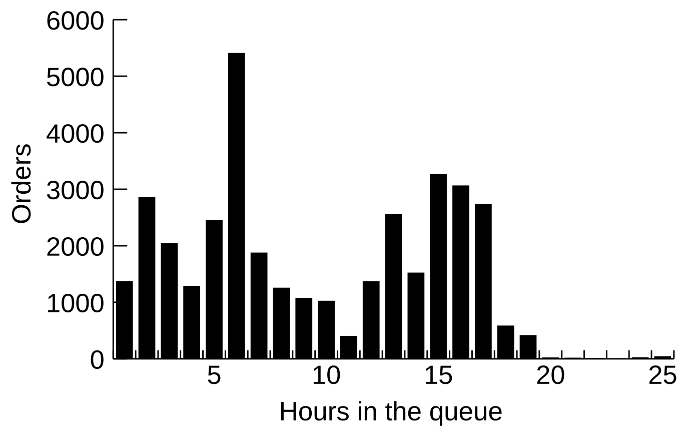

Figure 2: Average time spent in the induction queue for orders in September. The peaks correspond to the difference between arrival peak times and the times of the next picking waves.

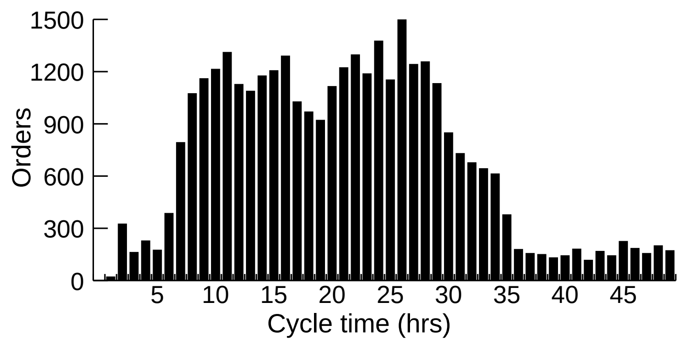

Figure 3: Average order cycle time (the sum of queue time and processing time) for September. Queue time peaks are still evident, although mitigated somewhat by processing time variance.

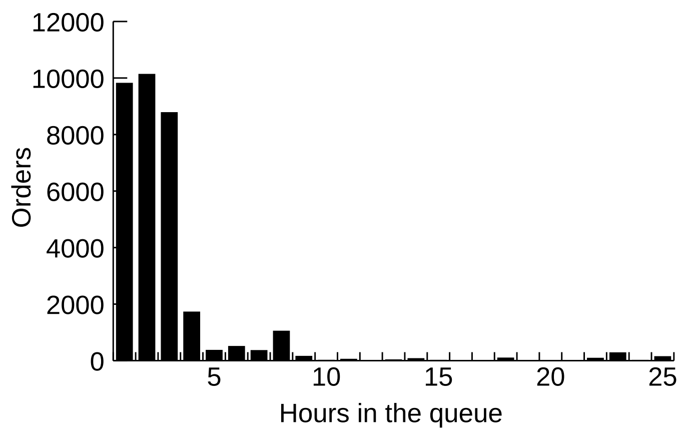

Figure 4: Average order queue time for November, after increasing the number of waves. Most orders begin processing within a few hours of arrival to the warehouse.

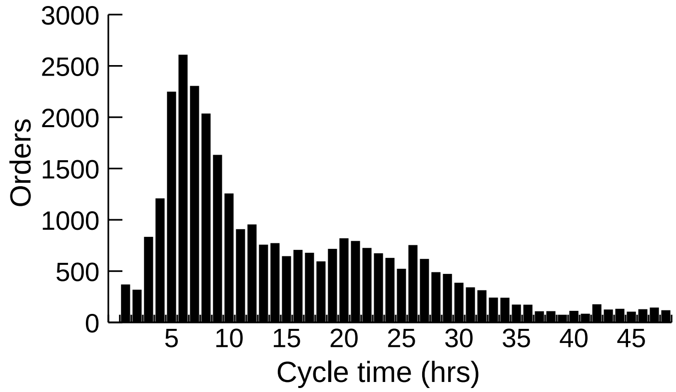

Figure 5: Average order cycle time for November, after increasing the number of waves from 2 to 4.

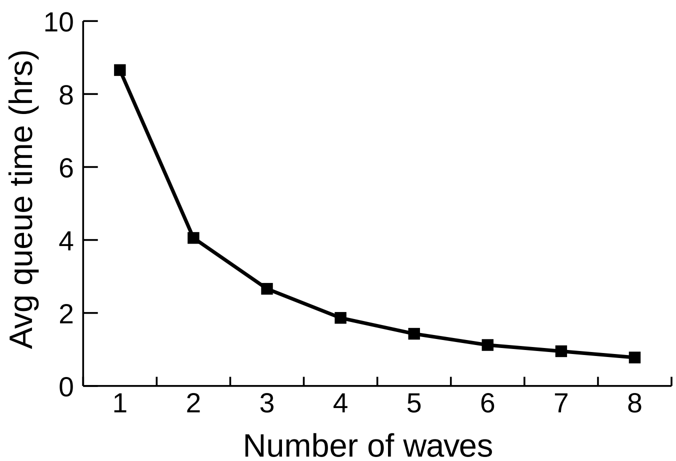

Figure 6: Average queue times for different numbers of picking waves, using data from the test site.

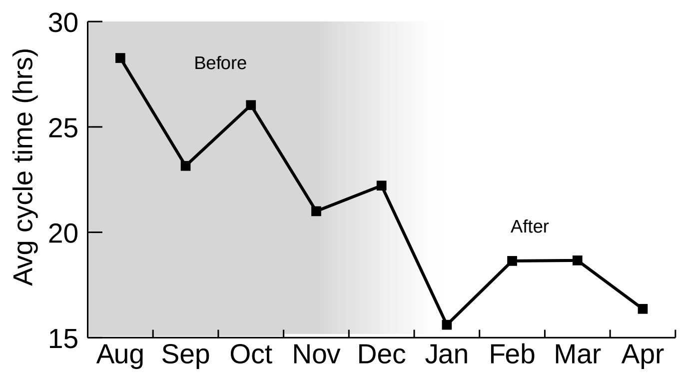

Figure 7: Average order cycle times before and after implementing the results of our model. Managers began making changes to the waves in November, eventually settling on 6 in January.